

# Optimized channel and delay selection for contention resolution in optical networks

W. Rogiest, K. De Turck, K. Laevens, D. Fiems, H. Bruneel, S. Wittevrongel  
SMACS Research Group · TELIN Department (IR07) · Ghent University (UGent)  
St.-Pietersnieuwstraat 41 · 9000 Ghent · Belgium · Wouter.Rogiest@UGent.be

**Abstract**—Both optical packet switching and optical burst switching provide viable alternatives to the current electronic switching in the backbone. To resolve contention, contiguous packets/bursts are sent over different channels, and/or provided with different Fiber Delay Line (FDL) buffer delays. Typically, the resulting channel and delay selection (CDS) algorithm bases its decision on the horizon value of the different channels. Up to now, a horizon algorithm known as MING (MINimal Gap) was assumed to provide minimal loss. Relying on Markov chain-based performance modeling and a Markov decision process (MDP), we reexamine the optimality of MING. This enables us to construct CDS algorithms that outperform MING, and this for any buffer size, traffic load and packet/burst size distribution. Further, we consider burst-size-dependent and load-dependent scheduling, two stochastic mechanisms that, together with preventive dropping, enhance the algorithm’s performance. The paper presents the developed method in detail, together with detailed results of the performance gain realized.

**Keywords**—optical packet switching, optical burst switching, optical buffer, FDL buffer, performance optimization, Markov decision process

## I. INTRODUCTION

The widespread interest in new web applications like High Definition on-demand video streaming is rapidly pushing the current network to its capacity limits. While the current backbone provides transport capacities of well beyond 10 Tbit/s per fiber, this capacity is only available for transmission from node to node. Current end-to-end communication suffers capacity loss from inflexible switching in intermediary nodes, urging for a more flexible approach to optical switching. Addressing this need, both optical burst switching (OBS) [1] and optical packet switching (OPS) [2] provide future-proof alternatives for the next-generation network.

Following a packet-based approach, OPS and OBS bring about a multiplexing gain by performing resource sharing for both switching and transmission. This however implies that packets/bursts may contend for the same channel at the same time, thus requiring a contention resolution scheme. Since random access memory (RAM) is not available in the optical domain, light is buffered by sending it through a fiber delay line (FDL). Since the number of available delay lines is limited to, say, 2 to 5 delay lines per node, the number of corresponding delays is also limited, and optimization of these delay values is crucial [3]. The resulting contention resolution scheme is a channel and delay selection (CDS) algorithm [4].

The core of the CDS problem is the discrete number of assignable delays. Due to this, on a given wavelength, it is in general not possible to schedule a packet/burst just after the transmission of a previous one. Therefore, on each outgoing wavelength, so-called *gaps* or *voids* occur, which are time periods during which no transmission takes place on the outgoing wavelength, even though packets/bursts are awaiting transmission in the optical buffer, leading to capacity loss.

In [5], a CDS algorithm minimal gap (MING) is introduced, which converts packets/bursts to the wavelength which results in the minimal gap size. Simulation results point out that MING outperforms classical delay-oriented algorithms like Join-The-Shortest-Queue, and therefore is assumed to minimize packet/burst loss [4]. In this paper, we show that not MING, but a different CDS algorithm realizes minimal loss in an optical packet/burst switch, and varies over different settings. Further, we examine several stochastic mechanisms that allow to further mitigate loss: preventive dropping [6], load-dependent scheduling and burst-size-dependent scheduling. Methodologically, we apply two complementary techniques: an analytic performance model based on general Markov chain theory, and an optimization method with a Markov Decision Process (MDP), a well-known tool for discrete-time optimization [7]. Note that we refer to packets/bursts merely as bursts below.

## II. SYSTEM DESCRIPTION

We assume a discrete-time setting, with time divided in time slots of fixed size. We number bursts in the order at which they arrive at the buffer, and denote by  $\{T_k\}$  an iid sequence of random variables representing the interarrival times, with discrete probability distribution function  $t(n)$ ,  $n \in \mathbb{N}_0$ , cumulative distribution  $F_T(n) = \sum_{i=1}^n t(i)$ ,  $n \in \mathbb{N}_0$  and expectation  $E[T_k] = \sum_{n=1}^{\infty} nt(n) < \infty$ . Similarly, the burst sizes form an iid random sequence  $\{B_k\}$  with discrete probability distribution  $b(n)$ ,  $n \in \mathbb{N}_0$ , and expectation  $E[B_k] = \sum_{n=1}^{B_M} nb(n)$ . Burst sizes are bounded by some value  $M$ , so that  $b(n) = 0$  if  $n > M$ .

We assume that the FDL set is general, or non-degenerate [8]. We denote the FDL set by  $\mathcal{A} = \{a_0, a_1, a_2 \dots a_N\}$  with  $a_0 = 0$ , and  $a_0 < a_1 < \dots a_N$ , with realizable delays all in this set. In several practical examples, we will consider a degenerate buffer setting. In that case, line lengths are equal to multiples of the granularity  $D$ ,  $a_i = iD$  for  $i = 0, 1 \dots N$ .

This buffer we assume located at the output interface of an optical burst (or packet) switch, and available for contention

resolution on two distinct wavelengths  $\lambda_1$  and  $\lambda_2$ . To enable two-wavelength contention resolution, we assume also that means for full wavelength conversion are present, together with a switching matrix, allowing to switch bursts to either  $\lambda_1$  or  $\lambda_2$ .

For further use, we introduce the notation

$$\lceil n \rceil_{\mathcal{A}} = \inf\{a_i \in \mathcal{A} : a_i \geq n\}, \quad n \in \mathbb{N}, \quad (1)$$

where  $\inf\{\}$  denotes infimum. The difference  $\lceil n \rceil_{\mathcal{A}} - n$  accounts for the gap or void created by an incoming burst. If this burst requests for a delay  $n$  and  $\lceil n \rceil_{\mathcal{A}} \leq a_N$ , the required delay can be realized; if not, the burst is dropped. On a given wavelength  $\lambda_1$  or  $\lambda_2$ , the scheduling horizon is defined as the earliest time at which all previous bursts will have left the system. The only state information to be kept for horizon scheduling is the scheduling horizon of the two wavelengths involved. However, to account for the possibility of burst-size-dependent scheduling, we add the burst size of the burst requiring scheduling, so obtaining a three-dimensional state space.

### III. EXACT PERFORMANCE MODEL

In the following, we present a system analysis in terms of transition probabilities of a two-dimensional Markov chain, allowing to calculate exact loss probability (LP) values.

#### A. Actions

The system description is in terms of the scheduling horizon, as seen by an arbitrary arrival  $k$  with burst size  $B_k$ . Associated with the two wavelengths  $\lambda_1$  and  $\lambda_2$  are the scheduling horizon values  $H_k^1$  and  $H_k^2$ , gathered in a two-dimensional scheduling horizon vector  $\mathbf{H}_k = (H_k^1, H_k^2)$ . Upon the arrival of burst  $k$ , wavelengths are indexed in order of increasing horizon value, such that  $H_k^1 \leq H_k^2$ . As such, the index  $i$  in  $H_k^i$  refers to the relative length of the horizon, and *not* to the index of the wavelength to which the horizon is associated when burst  $k$  arrives. The total state space vector  $\mathbf{S}_k$  is equal to the combination of the horizon vector  $\mathbf{H}_k$ , and the burst size of the burst that is to be scheduled,  $\mathbf{S}_k = (H_k^1, H_k^2, B_k)$ . The reason to include  $B_k$  in the state space is to enable burst-size-dependent scheduling. The process of burst arrival and transmission is governed by the CDS algorithm. More precisely, a scheduling algorithm in this context can be grasped by an action table, associating with each possible  $\mathbf{S}_k$  an action  $c_k$ . In this paper, we consider three actions:  $c_k = 1$ , consisting in choosing the wavelength with shortest horizon;  $c_k = 2$ , consisting in choosing the longest horizon; and  $c_k = 3$ , consisting in dropping the burst. Note that, if both horizon values exceed the maximum delay  $a_N$ , the only possible action is action 3, but that, on the other hand, it may be useful to perform action 3 also if this is not the case, which is the motivation of preventive drop, discussed below.

The actions  $\{1, 2, 3\}$  suffice to characterize any CDS algorithm considered, regardless whether or not stochastic mechanisms are used. As a CDS, MING chooses the horizon  $n$  with smallest  $\lceil n \rceil_{\mathcal{A}} - n$ . This can be formalized in an action table as given in Fig. 1a, also in the case of a degenerate buffer with

$N = 2$ ,  $\mathcal{A} = \{0, 5, 10\}$ , and  $B = 6$ . More formally, an action table or *policy* is given by an action table or *policy matrix*  $\mathbf{P}$ , conditioned on the state space  $(i, j, n)$  as seen upon arrival of an arbitrary burst  $k$ , with entries

$$p_{ijn} = \begin{cases} 1 \text{ or } 2 \text{ or } 3 & 0 \leq i \leq j \leq a_N \\ 1 \text{ or } 3 & i < a_N + 1 \leq j < a_N + B_M \\ 3 & a_N + 1 \leq i \leq j < a_N + B_M \end{cases}$$

and  $0 < n \leq B_M$ . If the possibility of preventive drop is excluded, the range of possibilities narrows to 1 or 2 if  $0 \leq i \leq j \leq a_N$ , and 1 if  $i < a_N + 1 \leq j < a_N + B_M$ .

Given this, it is insightful to consider the two-dimensional matrices  $\mathbf{P}_n$ , which describe the action table to be followed, given a system state  $\mathbf{S}_k = (i, j, n)$ , and conditioned on the assumption that a burst  $k$  of size  $B_k = n$  is to be scheduled. Each of these matrices can be split up in block matrices, as

$$\mathbf{P}_n = \left[ \begin{array}{c|c} \mathbf{P}_n^{123} & \mathbf{P}_n^{13} \\ \hline \mathbf{0} & \mathbf{P}_n^{33} \end{array} \right], \quad (2)$$

with the super-indices  $\{123, 13, 3\}$  referring to the actions available from that state *if* preventive drop is available. Matrices  $\mathbf{P}_n^{123}$  and  $\mathbf{P}_n^{33}$  are upper-triangular (the latter, with only “3” as entries), whereas  $\mathbf{P}_n^{13}$  is a dense stochastic matrix. This block structure can also be distinguished in the case of MING (see Fig. 1a), for which the scheduling is independent of the burst size  $B_k$ , and therefore,  $\mathbf{P}_1 = \mathbf{P}_2 = \dots \mathbf{P}_{B_M}$ . However, in the case of burst-size-dependent scheduling, all matrices ( $B_M$  in number) in general differ, so allowing for a more refined channel and delay selection. Finally, note that, in the case that no preventive drop is allowed,  $\mathbf{P}_n^{123}$  only contains actions 1 and 2, and  $\mathbf{P}_n^{13}$  only action 1.

#### B. Transition probabilities

For any action table assumed, a system state transition is initiated by the arrival of burst  $k$  with size  $B_k$ , and terminated by the arrival of burst  $k + 1$ ,  $T_k$  slots later. The associated Markov chain has a three-dimensional state space  $\mathbf{S}_k$ , and the transition matrices  $\mathbf{M}_h$ ,  $h \in 1, 2, 3$ , with probabilities  $m_h(l, m, p | i, j, n)$  describing the transition from  $\mathbf{S}_k$  to  $\mathbf{S}_{k+1}$ , defined as

$$\Pr[\mathbf{S}_{k+1} = (l, m, p) | \mathbf{S}_k = (i, j, n), c_k = h]$$

with  $0 \leq i \leq j < a_N + B_M$ ,  $0 \leq l \leq m < a_N + B_M$ ,  $1 \leq n \leq B_M$ ,  $1 \leq p \leq B_M$ ,  $h \in 1, 2, 3$ . Depending on the action taken, the transition probabilities take on a different form.

a)  $c_k = 1$ : In this case, regardless of the horizon  $j$  of wavelength 2, the horizon of wavelength 1 as seen upon arrival is sufficiently small,  $i \leq a_N$ , so that burst  $k$  can surely be buffered on the wavelength with horizon 1. Allocating burst  $k$  pushes horizon 1 to  $\lceil i \rceil_{\mathcal{A}} + n$  slots just after arrival, whereas horizon 2 remains unaltered, at a value of  $j$  slots. Associated is the (six-dimensional) transition matrix  $\mathbf{M}_1$ , with probabilities  $m_1$  that can be written as the sum of  $m_1^+$  and  $m_1^-$ , conditioning on whether the new scheduling horizon of  $\lambda_1$ ,  $\lceil i \rceil_{\mathcal{A}} + n$ , remains below  $j$ , or not, respectively. In case

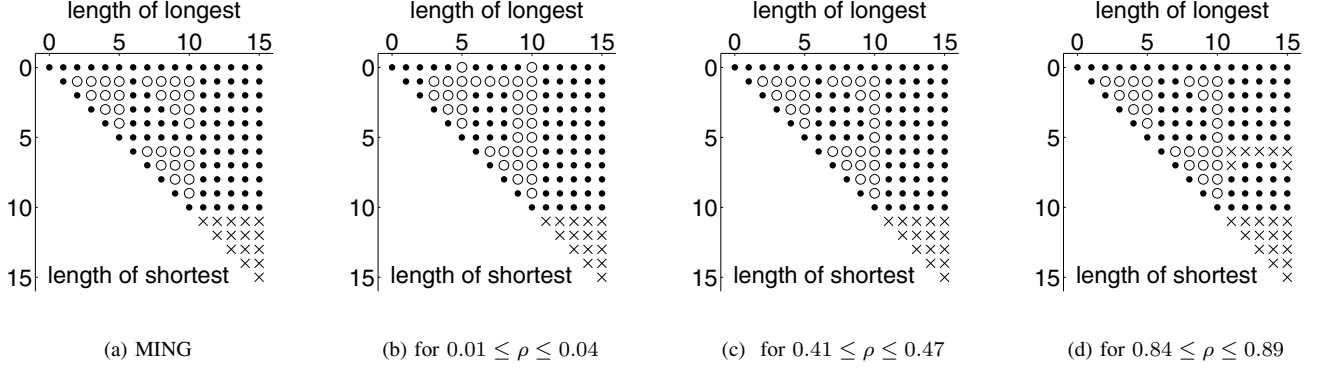


Fig. 1: A CDS algorithm connects an action to each pair of longest horizon (x-axis) and shortest horizon (y-axis). The actions  $\{1$  (join shortest),  $2$  (join longest),  $3$  (drop) $\}$  are indicated by  $\{\bullet, \circ, \times\}$ , respectively.

that  $\lceil i \rceil_{\mathcal{A}} + n \leq j$ , wavelength indexing remains unchanged for arrival  $k+1$ , and transitions are made with associated probability  $m_1^+(l, m, p | i, j, n)$  defined as

$$\Pr[\mathbf{S}_{k+1} = (l, m, p) | \mathbf{S}_k = (i, j, n), c_k = 1, \lceil i \rceil_{\mathcal{A}} + n \leq j].$$

Given the assumptions on inter-arrival and burst size distribution, the probabilities  $m_1^+(l, m, p | i, j, n)$  are obtained as

$$\begin{cases} t(q)b(p) & 1 \leq q < j \text{ and } l = \lceil i \rceil_{\mathcal{A}} + n - q^+ \\ & \text{and } m = j - q \\ (1 - F_T(j-1))b(p) & (l, m) = (0, 0) \end{cases}$$

and zero elsewhere. In the opposite case,  $\lceil i \rceil_{\mathcal{A}} + n > j$ , the index of the horizons is swapped in order to have  $H_{k+1}^1 \leq H_{k+1}^2$ . The associated transition probability is  $m_1^-(l, m, p | i, j, n)$  and represents

$$\Pr[\mathbf{S}_{k+1} = (l, m, p) | \mathbf{S}_k = (i, j, n), c_k = 1, \lceil i \rceil_{\mathcal{A}} + n > j].$$

Its values are

$$\begin{cases} t(q)b(p) & 1 \leq q < \lceil i \rceil_{\mathcal{A}} + n \\ & \text{and } l = [j - q]^+ \\ & \text{and } m = \lceil i \rceil_{\mathcal{A}} + n - q \\ (1 - F_T(\lceil i \rceil_{\mathcal{A}} + n - 1))b(p) & (l, m) = (0, 0) \end{cases} \quad (3)$$

and zero elsewhere.

b)  $c_k = 2$ : In this case, the horizon value  $j$  of wavelength 2 as seen upon arrival is sufficiently small,  $j \leq a_N$ , so that burst  $k$  can be buffered on the wavelength associated with horizon 2. Allocating burst  $k$  pushes the horizon 2 to  $\lceil j \rceil_{\mathcal{A}} + n$ , while horizon 1 remains at  $i$ . Since  $i \leq j$ , a fortiori,  $i \leq \lceil j \rceil_{\mathcal{A}} + B_k$ , and therefore, the index of the wavelengths is never switched in case of action 2, and the associated probability  $m_2$  entirely characterizes the corresponding transition, with the expression directly related to (3), through

$$m_2(l, m, p | i, j, n) = m_1^-(l, m, p | j, i, n).$$

c)  $c_k = 3$ : The buffer is found in blocking state, with both  $i$  and  $j$  larger than  $a_N$ . Since the burst size is upper-bounded by  $B_M$ , and the minimal inter-arrival time is assumed equal to 1, the scheduling horizon value is smaller than or equal to  $a_N + B_M - 1$ , resulting in  $a_N < i \leq j < a_N + B_M$ .

Action 3 corresponds to discarding arriving burst  $k$ . The scheduling horizon remains unaltered by the arrival, and the involved transition probabilities  $m_3(l, m, p | i, j, n)$  are as follows,

$$\begin{cases} t(q)b(p) & 1 \leq q < j \text{ and } l = [i - q]^+ \\ & \text{and } m = j - q \\ (1 - F_T(j-1))b(p) & (l, m) = (0, 0) \end{cases}$$

and zero elsewhere.

As such, all transition probabilities are known as soon as one assumes a certain CDS (and corresponding policy matrix  $\mathbf{P}$ ). From there, one can calculate the sparse transition matrix  $\mathbf{M}$  associated with them. From the obtained  $\mathbf{M}$ , using standard numerical means, one can extract the left eigenvector associated with eigenvalue 1, yielding the steady-state distribution. Denoting these probabilities by  $\Pr[H^1 = i, H^2 = j, B_k = n] = s(i, j, n)$ ,  $0 \leq i \leq j < a_N + B_M$ ,  $1 \leq n \leq B_M$ , one obtains the probability that an arbitrary arriving burst is lost by evaluating the probability that action 3 is taken. The (plain) loss probability (LP) (differing from the per-byte LP, see below) is thus obtained as

$$\text{LP} = \sum_{n=1}^{B_M} \sum_{i=0}^{a_N+B_M-1} \sum_{j=i}^{a_N+B_M-1} s(i, j, n) \delta_{p_{ijn}, 3},$$

where  $\delta_{p_{ijn}, 3}$  denotes the Kronecker delta, which equals one if  $p_{ijn} = 3$ , and zero elsewhere.

#### IV. MARKOV DECISION PROCESS

In the following, we apply an MDP technique to determine an optimal policy matrix  $\mathbf{P}$ , given a certain value of the traffic load  $\rho = \mathbb{E}[B_k]/(2\mathbb{E}[T_k])$ , in a way that is similar to the one described in [7], as it is based on the policy iteration algorithm described there. Given the state, one of three actions  $\{1, 2, 3\}$  should be chosen, so that the total reward is maximized. Each action constitutes a way to handle arriving bursts, and the choice for a given action is conditioned on the scheduling horizon as seen by the arriving bursts. Since we aim to minimize the loss probability, rewards are negative. Rather than optimizing on plain loss probability, we make rewards proportional to the size of the burst that is lost, namely  $-B_k$ ,

and assign a reward of zero to each accepted burst. This optimizes for minimal per-byte loss probability per byte, and so avoids the situation where an algorithm drops large bursts “just because of their size”. This reward function trivially maps on the set of actions: action 1 and 2 correspond to zero reward, action 3 yields a reward of  $-B_k$ .

The policy iteration algorithm now consists in choosing an arbitrary initial (three-dimensional) policy array  $\mathbf{P}$ , with an arbitrary choice among the actions possible in a given state. In the case that preventive drop is allowed, conditioned on burst size  $n$ , the allowed actions are 1,2 or 3 for  $\mathbf{P}_n^{123}$ , 1 or 3 for  $\mathbf{P}_n^{13}$ , and 3 for  $\mathbf{P}_n^3$ . In the case no preventive drop is allowed, the allowed actions are 1,2 for  $\mathbf{P}_n^{123}$ , 1 for  $\mathbf{P}_n^{13}$ , and 3 for  $\mathbf{P}_n^3$ .

Given the (three-dimensional) policy  $\mathbf{P}$ , a value is determined for each state  $\mathbf{S} = (i, j, n)$ , consisting of the immediate reward, and all rewards to be earned in the future, taking into account the possible state evolution (as dictated by the probabilities  $m_1^+$ ,  $m_1^-$ ,  $m_2$  and  $m_3$  obtained in Sect. III). Then, each policy iteration consists in (i) determining the new policy  $\mathbf{P}'$  that maximizes the expected reward, given the computed values of the previous steps and the probabilities  $m_1^+$ ,  $m_1^-$ ,  $m_2$  and  $m_3$ ; and next (ii) computing the new values, given the new policy  $\mathbf{P}'$ . The policy is reiterated until no change takes place in the policy in step (i). We refer to [7] for further details.

## V. OPTIMIZATION EXAMPLES

In this section, we apply the optimization techniques developed in the previous sections. We assume a Bernoulli arrival process, which is the discrete-time counterpart of a Poisson arrival process. This arrival process is often-assumed and appropriate for performance modeling of backbone network traffic [9]. At the beginning of each slot, either one or no arrival occurs, with probability  $p$  or  $1 - p$ , respectively. Given this, the inter-arrival time  $T_k$  between the arrival of burst  $k$  and burst  $k+1$  (as introduced in general in (II)) follows a geometric distribution, with probability density function  $t(n) = p \cdot \bar{p}^{n-1}$ ,  $n \in \mathbb{N}_0$ , and cumulative distribution function  $F_T(n) = 1 - \bar{p}^n$ ,  $n \in \mathbb{N}_0$ , and expected value  $E[T_k] = 1/p$ . In the below, different values of the traffic load  $\rho = pE[B_k]/c$  (with  $c$  the number of wavelengths,  $c = 2$ ) are considered; the difference is obtained by varying  $p$ , with the burst size distribution unaltered.

We first assume burst sizes fixed to some integer value of  $B$  slots, and corresponding burst size distribution  $b(n) = \delta_{n,B}$ , (where  $\delta_{i,j}$  again denotes the Kronecker delta),  $B_M = B$ , and  $E[B_k] = B$ . This distribution, combined with geometric inter-arrival times, is an often-studied combination in case of single-wavelength optical buffers. Previous studies revealed that a degenerate buffer setting, with  $D = B - 1$  is almost always (but not always, see [6]) the optimal choice if the load  $\rho$  remains below some threshold load  $\rho_{th} \approx 0.6$  [8]. For multiple wavelengths, fewer results are available; for a degenerate setting however,  $D = B - 1$  is also an advantageous choice in terms of performance, see [10]. As such, we also consider this choice here, in several specific setting of modest

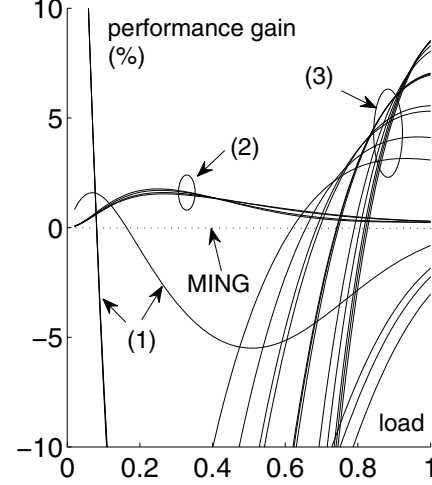


Fig. 2: Performance gain (in %) of the obtained CDS algorithms over MING.

numerical complexity, and  $B = 6$  slots. Using the formulas of the previous sections, for each load value  $\rho = i \cdot 0.01$ ,  $i = 1 \dots 100$ , we performed an independent MDP optimization, each yielding a separate policy matrix  $\mathbf{P}$ , optimized for minimal loss at exactly the load assumed. Given the limited set of states,  $(ND + B - 1)(ND + B)/2 = 136$ , each with a maximum of three possible actions (or less), it comes as no surprise that several load values yielded exactly the same policy as optimal policy.

First, we consider the case of  $N = 2$  and  $D = 5$  (and thus,  $\mathcal{A} = \{0, 5, 10\}$ ), and a basic optimization, with the possibility of preventive drop. The MDP optimization was performed independently for each load value  $\rho = i \cdot 0.01$ ,  $i = 1 \dots 100$ , and yielded over 20 different CDS algorithms, each valid for certain intervals of the load  $\rho$ . Three of these algorithms are displayed on Fig. 1: Fig. 1b ( $0.01 \leq \rho \leq 0.04$ ), Fig. 1c  $0.41 \leq \rho \leq 0.47$ , and Fig. 1d ( $0.84 \leq \rho \leq 0.89$ ). The load intervals fall apart in three groups, corresponding to low traffic intensity (1), with  $0.01 \leq \rho \leq 0.11$ , intermediate traffic intensity (2), with  $0.12 \leq \rho \leq 0.65$ , and high traffic intensity, with  $0.66 \leq \rho \leq 1.00$ . The rationale of this grouping can be best understood when considered along with the performance output, as it can be obtained exactly, by using the CDS algorithms as input to the exact model of Sect. III. In Fig. 2, the performance of the algorithms is compared in an exact manner to that of MING (Fig. 1a), in terms of loss probability reduction. Clearly, the CDS algorithms of group (2) outperform MING for any value of the load, but the performance gain is small (1–2%), because MING is close to optimal for such intermediate traffic intensity. This contrasts with the situation for low load, for which the algorithms of group (1) realize a performance gain of over 30%. One of the algorithms (the one optimal for  $0.01 \leq \rho \leq 0.04$ ) is displayed in Fig. 1b. Further, note that part of the curve for (1) is not shown in Fig. 2 to avoid improper scaling; also note that for  $\rho = 0.01$ , the performance gain rises to 37.9%. Indeed, the

$\rho$	0.20	0.40	0.60	0.80	1.00
(a)	5.36	2.92	1.49	6.31	17.86
(b)	44.65	21.00	11.86	5.59	1.70

TABLE I: Loss reduction (in %) of an optimized load-dependent CDS algorithm over MING, for 4 lines ( $N = 4$ ) and (a)  $\mathcal{A} = \{0, 5, 10, 15, 20\}$  with fixed burst sizes and preventive drop, and (b)  $\mathcal{A} = \{0, 6, 10, 16, 20\}$  with varying burst sizes and without preventive drop.

results for group (1) would have been hard to obtain with sufficient accuracy through simulation, whereas they can be calculated instantly and exactly with our method.

Reconsidering the case of  $N = 2$  and  $D = 5$  ( $\mathcal{A} = \{0, 5, 10\}$ ) without the possibility of preventive drop yields very similar results: all the curves of group (1) and (2) in Fig. 2 are repeated, whereas the curves of group (3) do not occur (since the algorithms of this group involve preventive drop). The algorithms of group (3) profit from the “speculative benefit” of preventive drop. This benefit comes about for high load situations, dropping those bursts that would cause large gaps, in order to be able to accommodate future bursts with (potentially) smaller gaps.

Finally, as introduced in Sect. II, the stochastic mechanism of load-dependent scheduling enables to reduce loss probability even further. Assumed that we have an accurate estimation of the current traffic load, and schedule bursts accordingly, one can obtain an improved loss performance curve, taking the maximum of all curves displayed in Fig. 2. While this is the ideal case, it should suffice to dynamically switch between three algorithms, in order to approach minimal loss performance rather close. The three algorithms are drawn from group (1), (2) and (3), in order to deal with low, medium and high traffic load, respectively. Note that such a dynamic algorithm can be used in practice on the same hardware as MING, by adapting the action tables of the CDS algorithm to the currently measured traffic load. The implementation aspects of such setting however go beyond the scope of this contribution.

To verify whether the optimization for deterministic burst size distribution for  $N = 2$  is representative also for larger FDL sets, we performed the same calculations for the case of  $N = 4$ ,  $B = 6$ ,  $D = 5$  and  $\mathcal{A} = \{0, 5, 10, 15, 20\}$ . Apart from being similar, the optimization space is larger, since the set of possible actions is larger. With preventive drop allowed, this resulted in about 50 different algorithms. The associated performance gain over MING in a load-dependent scenario (with preventive drop allowed), is displayed in Table I (a). For any load, more performance gain over MING is realized than in the case of  $N = 2$ .

Finally, we also consider the case of varying burst sizes, for which the number of possible optimal scheduling algorithms increases spectacularly, as the possibility of burst-size-dependent allows for further refinement of the CDS algorithm, especially when considered in combination with preventive drop and load-dependent scheduling. The specific setting we assume has a burst size distribution

$$b(n) = 0.5 \cdot \delta_{n, B_1} + 0.5 \cdot \delta_{n, B_2},$$

(with  $\delta_{i,j}$  again the Kronecker delta),  $B_1 < B_2$ ,  $B_M = B_2$ , and  $E[B_k] = (B_1 + B_2)/2$ . We assume  $B_1 = 5$  and  $B_2 = 7$ , and a non-degenerate FDL set  $\mathcal{A} = \{0, 6, 10, 16, 20\}$  with thus  $N = 4$ .

Burst-size-dependent optimization without preventive drop yields the results listed in Table I (b). It illustrates that burst-size-dependent scheduling performs much better than MING in a non-degenerate setting, especially for low traffic loads, with reductions of 20–40%. Note that the introduction of preventive drop (not shown here) also enables to realize loss reduction for high traffic load, due to the “speculative benefits” mentioned earlier.

## VI. CONCLUSIONS

In this contribution, a CDS performance model and optimization method were presented, allowing to reexamine the optimality of MING. By means of an MDP, we constructed CDS algorithms that outperform MING. Several CDS algorithms outperform MING for any value of the load, even when applied statically, without taking the load into account. By combining the obtained policies in a load-dependent algorithm, overall performance is improved further, especially when combined with preventive drop. Finally, also burst-size-dependent scheduling was examined, allowing a loss reduction of 20–40% for the given setting.

Since the results presented here only apply to a limited setting (two wavelengths), current research efforts focus on the extension to a general number of wavelengths. For such setting, the presented model inevitably suffers state space explosion; therefore, a main research goal is to come up with appropriate heuristics and approximations.

## REFERENCES

- [1] C. Qiao and M. Yoo. Optical burst switching—a new paradigm for an optical internet. *Journal on High-Speed Networks*, 8:69–84, 1999.
- [2] S. Yao, B. Mukherjee, and S. Dixit. Advances in photonic packet switching: An overview. *IEEE Communications Magazine*, 38(2):84–94, 2000.
- [3] A. Gadkar and S. Subramaniam. Optimal FDL design for time-wavelength crossconnects and optical packet switches. In *Proceedings of the IEEE International Conference of Communications 2009 (ICC)*, pages 1–5. IEEE, 2009.
- [4] F. Callegati, W. Cerroni, and G. S. Pavani. Key parameters for contention resolution in multi-fiber optical burst/packet switching nodes. *Proceedings of the Fourth IEEE International Conference on Broadband Communications, Networks and Systems, Broadnets 2007 (Raleigh)*, pages 217–223, 2007.
- [5] F. Callegati, W. Cerroni, and G. Corazza. Optimization of wavelength allocation in WDM optical buffers. *Optical Networks Magazine*, 2(6):66–72, 2001.
- [6] J. Lambert, B. Van Houdt, and C. Blondia. Single-wavelength optical buffers: Non-equidistant structures and preventive drop mechanisms. *Proceedings of the 2005 Networking and Electronic Commerce Research Conference, NAEC 2005 (Riva del Garda)*, pages 545–555, 2005.
- [7] H.C. Tijms. *Stochastic Modelling and Analysis: A Computational Approach*. J. Wiley and Sons, 1986.
- [8] W. Rogiest, J. Lambert, D. Fiems, B. Van Houdt, H. Bruneel, and C. Blondia. A unified model for Synchronous and Asynchronous FDL Buffers allowing Closed-Form Solution. *Performance Evaluation*, 66(7):343–355, 2009.
- [9] J. Cao and K. Ramanan. A poisson limit for buffer overflow probabilities. In *Proceedings Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 994–1003. IEEE, 2002.
- [10] W. Rogiest, K. Laevens, D. Fiems, and H. Bruneel. Modeling the Performance of FDL Buffers with Wavelength Conversion. *IEEE Transactions on Communications*, 57(12):3703–3711, 2009.